PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-301935
(43)Date of publication of application : 13.11.1998

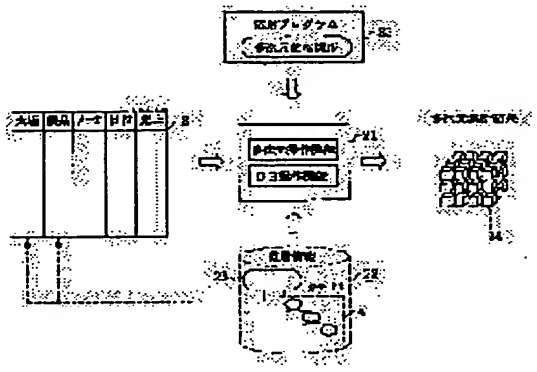
(51)Int.Cl. G06F 17/30

(21)Application number : 09-104830
(22)Date of filing : 22.04.1997

(71)Applicant : MITSUBISHI ELECTRIC CORP
(72)Inventor : IWASAKI HIROFUMI
ANDOU TAKAAKI

(54) DATA PROCESSING METHOD

(57)Abstract:
PROBLEM TO BE SOLVED: To provide a data processing method capable of eliminating the necessity of preprocessing such as the generation of an index and easily executing multi-dimensional operation even in the case of executing multi-dimensional operation of the data of a detailed level.
SOLUTION: Data related to 'branch' and 'product' included in data existing in a real table 3 are obtained based on the category of hierarchical information 22 and added to the table 3 as a virtual string 23. Then a multi-dimensional totalization result 24 totalizing the contents of all totalization items in the table 3 is obtained by executing the sort post-totalization of respective totalization items in the table 3 as sorts.



LEGAL STATUS

[Date of request for examination] 22.04.1997
[Date of sending the examiner's decision of rejection] 21.12.1999
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

CLAIMS

[Claim(s)]
[Claim 1] It has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which searches the summing field data of the upper layer or a lower layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, The data-processing approach characterized by sorting the above-mentioned detail data and carrying out total processing by using the obtained summing field data as a key at the above-mentioned summing field data unit.
[Claim 2] It has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which totals the above-mentioned detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data,

DETAILED DESCRIPTION

[Detailed Description of the Invention]
[0001]
[Field of the Invention] Especially this invention relates to amelioration of the data-processing approach of being suitable for query processing of a multi dimensional database managerial system, and total processing, about the data-processing approach.
[0002]
[Description of the Prior Art] Generally, in order to pull out multi-dimension data at a high speed, an index is used for what is called a multi dimensional database, and it is performing data processing. There are the following two kinds also in the approach using an index. The multi dimensional database which generates internally the data which unlike the relational database holding the data of description level load beforehand the data of minimum total level required of multi-dimension actuation from a relational database etc., and are used by multi-dimension actuation based on these data. To the data which such a multi dimensional database holds, an index is generated simultaneously at the time of loading of data, or an internal generate time, and the inquiry of multi-dimension actuation is performed at a high speed.
[0003] Moreover, the multi dimensional database which holds the data of detail level like the usual relational database, extends relational database actuation, and performs multi-dimension actuation. In such a multi dimensional database, the index needed by the join used by multi-dimension actuation is generated beforehand, and the inquiry of multi-dimension actuation is performed at a high speed. There are two kinds of approaches described above.
[0004] In order to pull out the conventional multi-dimension data which were described above at a high speed, there is an approach indicated by JP,4-229371,A as an example which uses an index and is performing data processing. Although changing and searching the hierarchy level of provisions of classification is indicated, the provisions-of-classification list which is equivalent to an index as a retrieval means is required of this example.
[0005]
[Problem(s) to be Solved by the Invention] Since the conventional data-processing approach is constituted as mentioned above, to perform multi-dimension actuation from the data of detail level, it is necessary to generate an index as pretreatment. In the case of the former which loads beforehand the data of minimum total level still more nearly required of multi-dimension actuation from a relational database etc., it is necessary to load the data of total level from the data of detail level. When detail data new every day are added to a large quantity, these pretreatments must pretreat each time and have the problem that become very heavy processing, become weekly and monthly processing actually, and the sex is lost instancy, in a mass multi dimensional database.
[0006] Also when it was made in order to cancel the above troubles, and performing multi-dimension actuation to the data of detail level, this invention makes pretreatment of generating an index unnecessary, and aims at offering a multi-dimension actuation function easily.
[0007]
[Means for Solving the Problem] The data-processing approach concerning this invention is equipped with the data file by sequential organization of the detail data which consist of two or

more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. By using the step which searches the summing field data of the upper layer or a lower layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, and the obtained summing field data as a key, the above-mentioned detail data are sorted and total processing is carried out at the above-mentioned summing field data unit.

[0008] Moreover, it has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which totals the above-mentioned detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data, By using the step which searches the summing field data of the upper layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, and the obtained summing field data as a key, the above-mentioned total data are sorted and total processing is carried out at the above-mentioned summing field data unit.

[0009] Furthermore, it has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which totals the above-mentioned detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data, The step which searches the summing field data of the upper layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, By using the step and the above-mentioned delimiter which insert the obtained summing field data and its delimiter in the above-mentioned detail data as a key, the above-mentioned total data are sorted and total processing is carried out at the above-mentioned summing field data unit.

[0010] Moreover, it has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which totals the above-mentioned detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data, The step which chooses from the above-mentioned summing field data the summing field data made applicable to a total, The step which searches the summing field data of the upper layer or a lower layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, By using the step and the above-mentioned delimiter which insert the obtained summing field data and its delimiter in the above-mentioned detail data as a key, the above-mentioned total data are sorted and total processing is carried out at the above-mentioned summing field data unit.

[0011] The order relation between the level name for every hierarchy, the element names belonging to the level, and those elements is included in the above-mentioned layered structure information table further again.

[0012] Moreover, it totals for every combination of all containing the above-mentioned summing field data according the above-mentioned detail data contained in the element arranged by the sequence set up beforehand to selection of arbitration.

[0013] Furthermore, when the summary value by the above-mentioned total processing is zero, zero data are held as a summary value.

[0014]

[Embodiment of the Invention]

The gestalt 1 of operation of this invention is explained to a detail based on a drawing below gestalt 1. of operation. Completion of install and a setup of the detail database (Detail DB is called below) system in the gestalt 1 of this operation generates one environment shown in drawing 1 . In drawing 1 , 1 is a catalog and it is one set of PC. Only one environment can be

generated in Server and a catalog 1 exists in the environment one **. Under a catalog 1, it is possible to generate and delete two or more schemas 2. All the information about the schema 2 grade generated under a catalog 1 is managed by the above-mentioned detail DB system as catalog 1 information.
[0015] To a schema 2, it is possible to generate and delete two or more real tables 3. In a schema 2, it is SET. A default schema can be set up by SCHEMA. When not setting up an explicit default schema, the schema decided by the system turns into a default schema.
[0016] To each schema 2, it is possible to generate and delete the category 4 as a layered structure information table. In a category 4, the sequence between the elements (actual-key value included in the real table 3) in data-hierarchy structure and each hierarchy is expressed, and it consists of level of N individual to which the identifier was attached. It has a list of the element belonging to the level, and the order relation between elements in each level. Moreover, between level, it has the relation for every element and this expresses hierarchical relationship. Whether a train is seen on the level of which category 4 throat sets up in the category function on the original database language in a detail DB system. Drawing 2 is an example which shows the structure of a category 4, and 11 shows "level", such as a section and a name, and the "sequence" of each element [in / 12 / "elements", such as the Administrative Division and the personnel division, and 13 and / in 14 / the same level]. [the "relation" between each element]
[0017] The real table 3 is a sequential file in the fixed length who consists of two or more trains 5 (field). The table which can be defined as a detail DB system is only the real table 3. However, the "category function" which realizes the imagination train 5 which can change the view from the program of each train free is set up.
[0018] By applying a "category function" to a train 5, it becomes possible to access a real table in the imagination train 5. Only by defining the view (view) of a train 5 through a category 4, the treatment of each train 5 can be changed for every application program. It is made to look as if the train 5 which is not held directly existed on the real table 3 by specifically applying the above-mentioned category 4 to the train 5 which constitutes the real table 3 using a category function. Thereby, modification of the list of a key, selection of total level, etc. are attained.
[0019] Next, it explains that processing of the whole multi-dimensional processing flows. In many cases, what is called a dimension in the world of a multi dimensional database holds hierarchical structure, and various retrieval and totals on hierarchy level are possible for it. For example, if a personal name is made into an example, when the individual belongs to the firm, if an identifier is decided, it will be decided that the section and the section to which the man belongs will be a meaning. That is, if the layered structure of this personal name is independently defined even when only the personal name is held in the train 5 of the real table 3, the retrieval and total processing which used as the key the section which is a hierarchy's upper level, and a section will be attained. In the detail DB system in the gestalt 1 of this operation, these actuation can be performed to the real table 3 of the detail base. In order to realize the same function as the dimension in such a multi dimensional database especially, the view of the real table 3 which is raw data is changeable free by introducing the category 4 which is the layered structure of an imagination train.
[0020] Drawing 3 shows the outline of multi-dimensional processing, 3 shall be a real table and only the summing field of a "manufacturer", a "date", and "sales" should exist at the beginning. Next, the demand made applicable to a total also about a "branch office" and a "product" occurs, and based on the category 4 of a hierarchy 22, the imagination train 23 is processed as if it existed in the real table 3. Next, the multi-dimension total result 24 of having totaled about all the summing fields of the real table 3 is obtained by totaling after a sort as a sort key about each summing field of the real table 3, respectively. All the above actuation is controlled by the application program 25.
[0021] The train 5 which constitutes the real table 3 is roughly divided, and is classified into two. One is a train holding the information for identifying an identifier of an object which is expressed with a master code. As an example, although a personnel code, an identifier, etc. are mentioned, these are used, in order that both may identify a personal name. The second is a train holding

the information for identifying a condition of an object which is expressed numerically. A selling price, the sale number, etc. are mentioned as this example. Although these former was called the "dimension", the latter was called the "variable" and treatment is clearly divided in the general multi dimensional database, both these are intermingled in one line in the common world and the common detail DB system of SQL. In addition, it is GROUP when performing total processing by GROUPBY in a detail DB system. The train which the train specified as the key of BY specifies as a dimension and a set function corresponds to a variable.

[0022] Next, detail explanation is given about a category 4. It becomes possible to acquire the same effectiveness as the "dimension" in the many dimensions DB by applying a category 4 to the train 5 with the real table 3. The following information is included in the category.

(1) In case the concrete value category name of the special values ("ALL" etc.) of related list (4) each level of the element between the element list belonging to the level name for every category name (2) hierarchy and its level and the order relation (3) level between elements and a level name apply a category function, use them, the category name is unique within a schema, and if the level name is unique within a category, it is good. The element 12 of each level is an item used as the key in retrieval or a total, and can define sequence between the elements of the same level. However, this sequence is the case where have semantics only when ROLLUP and a CUBE total are performed, and it does not total, and simple GROUP. In BY, the sequence for every usual data type is applied. Thereby, setting out becomes possible for every user about the order of an output as a result of the below-mentioned roll-up total.

[0023] The layered structure as a category 4 is defined by defining relation with the element 12 of the level of one high order at the time of the element definition of each level. Retrieval and a total with an upper level are attained by defining this relation. It is also possible to prepare the category for oneself and to define only the required element 12 as an element 12 of level positively, when what he should process in the element 12 of a train 5 as an approach using this function is limited.

[0024] Although the special value "ALL" exists logically in a category 4, this is used as a value showing the "intermediate total" and the "major total" at the time of performing a roll-up and a cube total. Drawing 4 is what showed matching with the real table 3 and a category 4, and since Taro Yamada in the real table 3 exists in a category 4, he can match as it is.

[0025] In the detail DB system in the gestalt 1 of this operation, the category function which realizes the imagination train which does not exist actually is offered to the usual train which constitutes the real table 3. In order to use a category function, after defining the above-mentioned category 4, it is required to make the level of an either a train 5 or the category 4 map to the real table 3 which applies it. Since it is possible to define two or more layered structures by changing a view to one train 5, the relation between a category 4 and a train 5 is set to N to 1. This shows that two or more completely different categories like an "organization" and an "executive" are applicable to the train of an identifier. Moreover, it is possible to define two or more level in one category 4, and a category function can be applied to each of that level. This can define two or more level called the "section", a "section", and "name" in a category called an organization, and shows that a category function is applicable on every level in it.

[0026] The sequence 14 between the elements 12 defined within the category 4 is effective only when ROLLUP and a CUBE total are performed to the train acquired with the application of the category function. Moreover, it not only maps a value to the element 12 of each level in a category 4, but a category function can be mapped to the subelement of each level.

[0027] Drawing 5 is what showed relation with the imagination train acquired by applying a train and a category, and by applying the category function to the organization (section) from an identifier shows notionally that the imagination train of a section name is generated and the imagination train of a section name is generated by applying the category function to the organization (section) from an identifier.

[0028] Drawing 6 is what showed the relation between a train and an imagination train concretely, and shows signs that the identifier on a train is generated as a section name on an imagination train by applying the category function to the organization (section) from an identifier.

[0029] The data type offered by the detail DB system in the gestalt 1 of this operation offers a "DATE mold" with the function similar to the view of a category. This DATE mold is the data type showing the date. Drawing 7 shows the layered structure of a DATE mold, and shows the layered structure by the tacit relation by the established practice from the former.
[0030] As the approach of gestalt 2. many dimensions total processing operation, they are ROLLUP and CUBE. Although mentioned, the gestalt 2 of the operation which applied this invention is explained to these based on drawing. GROUP in the database language of a detail DB system the total processing which used two or more specified keys as the 1 lump when two or more keys were specified and grouping was performed in BY — only carrying out (subtotal) — it is . If either of two or more specified keys takes the 1 surroundings (key breaking) by the total of ROLLUP or CUBE to it, total processing to there will be totaled further (intermediate total and major total), and it will return as a result. The difference in the output of ROLLUP and CUBE is shown in drawing 8 . "*" in drawing means all the elements contained in a train, and "ALL" means the sum total of all the elements of the train. The example of an expression of drawing 8 shows a thing without hierarchical relationship according to a three dimension (the key of a GROUP BY phrase is three pieces).
[0031] In order to consider the concrete example of ROLLUP and CUBE, the real table 3 with the category 4 shown in drawing 9 is considered. Moreover, the structure of the real table 3 is constituted as a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which shows drawing 10 and totals. Moreover, SQL is used as a database language.
[0032] GROUP shown in drawing 11 in order to obtain "the monthly field option raising table from January of the Katase branch office to March" to the real table 3 shown in drawing 10 The result obtained when the sentence which specifies retrieval of three kinds of detail databases, BY, ROLLUP, and CUBE, is published is the chart shown in drawing 12 . In drawing 12 (a) GROUP The total by BY, a total according [(b)] to ROLLUP, and (c) show the total result by CUBE, and television, video, a washer, a refrigerator, AV, household electric appliances, and the point that each totalizer of the sales in January – March is totaled are the descriptions in 1Q column in (c). In addition, "CATEGORY-" shown in each SQL sentence of drawing 11 shows a category function, and relating with the upper layer by the category 4 is performed.
[0033] Here, the art for asking for the roll-up and cube which are many dimensions-actuation is explained. The approach of asking for a roll-up focusing on sorting application is GROUP of SQL focusing on sorting application fundamentally. After resembling the processing which performs BY and sorting input data, a roll-up total can be obtained by performing total processing for every key breaking. In order to ask for the cube of N dimension focusing on sorting application, roll-up total processing can be obtained by repeating N times.
[0034] First, if direct roll-up total processing is performed to detail data, since a total overhead may become large, depending on the number of cases of detail data, it is the usual GROUP as a first stage story. It will ask for BY total. In drawing 12 , it has totaled by (a) for every possible combination of all containing television, video, a washer, a refrigerator, AV, household electric appliances, and all the summing field data of the moon. This GROUP It is totaled by a field and every month as it is shown in (b), when the roll-up total to a field and the moon is performed as a second stage story to the data (or detail data) which carried out BY. In such roll-up actuation, when same actuation is performed for every dimension, all cube total results will be obtained in a culmination.
[0035] Here, when a summary value is zero, zero data are held as a summary value by making a summary value into zero. In drawing 13 , when a **** branch office and the Imaizumi branch office are sale zero, the zero total which adds the data of sale zero and totals can be carried out. This zero summarize function can be carried out by assignment, and can choose ****.
[0036] Gestalt 3. this invention of operation is explained as a gestalt 3 of operation about the case where it applies to a pipeline merge sorter. Drawing 14 shows a part for the body of the data flow when performing the total by CUBE in a pipeline merge sorter. Setting to drawing, 31 is GROUP. It is what chose from data after performing the total by BY the element made applicable to a total, and stored those summing field data in the disk buffer (not shown), and this data reads

the data of two or more affairs into a pipeline merge sorter (not shown) inner buffer at once by the DMA transfer. Projection processing is performed in a pipeline merge sorter, and it will be in the condition which the content of the record is extended to sort key data (left-hand side in a buffer), and the data (right-hand side in a buffer) returned to a user, and shows in 32. Next, the sequence key for a sort and high order hierarchy data as a delimiter will be in the condition of the buffer which it is written in in a buffer and shown in 33 with the firmware (F/W) in a pipeline merge sorter, making a category 4 reference in the clearance left with the projection at the time of a DMA transfer. If all buffers are buried, data will be sent to the sort section 34 and will be sorted in order of the sort key on the left-hand side of a buffer. When lower layer data are written in in a buffer from a category 4, the content of processing is the same only by the location written in turning into a different location from the condition which shows in 33.

[0037] The result sorted in order of the sort key on the left-hand side of a buffer will be in the condition of the buffer shown in 35. Although total processing is performed by F/W, the total (roll-up total) by key breaking is performed here again. A total in a product and a field is completed by finally returning the content of the record of the right-hand side in a buffer to a user. Cube processing is completed by performing about all the elements that chose the above processing as the date, a branch office, and a manufacturer's object for a total.

[0038] Next, actuation of the gestalt 3 of operation is explained. Drawing 15 is GROUP which is a processing flow explaining actuation, totals for every possible combination of all that contains all summing field data from the detail data of the real table 3 first, and creates total data. BY is performed (step S1). Next, the summing field data made applicable to a total are chosen from summing field data, and it stores in a disk buffer (step S2). Since the summing field data which were not chosen are deleted, the record length will be in the condition of having been reduced. Here, there are few elements, it is possible also for storing all summing field data in a disk buffer, without choosing, does not need to search lower layer data from a category 4 in this case, and serves as retrieval of only upper layer data.

[0039] Next, the tooth space which extends the content of the record to the data returned to sort key data and a user by projection processing, and inserts the summing field data and the sequence key of the upper layer of a category or a lower layer is set up (step S3). A category is searched and the summing field data and its sequence key of the upper layer or a lower layer are inserted in the above-mentioned tooth space (step S4). A sequence key is sorted as a sort key (step S5).

[0040] Next, it totals about the summing field data made into the object of a sort key by the set operation, and the line for a total is added (step S6). Here, it judges whether total processing was ended about all the summing field data of the selected element (step S7), and when having not ended, the sequence key of the summing field data which carry out total processing is stored in step S3 at a disk buffer at return and a degree. When it ends, CUBE about the summing field data moved and chosen as step S8 is completed.

[0041] Sorting application is more possible at a high speed by not using the comparatively long alphabetic data of a data length as a sort key, but sorting the sequence key which is a code in a figure as a sort key as above-mentioned.

[0042]

[Effect of the Invention] Since this invention is constituted as explained above, it does effectiveness as taken below so.

[0043] Since it has the layered structure information table which set up the layered structure between summing field data beforehand, the summary value about the summing field data of the upper layer or a lower layer can be obtained from summing field data.

[0044] Moreover, since the step which totals detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data was prepared, the summary value about the summing field data of the upper layer can be obtained more at a high speed.

[0045] Furthermore, since the step which inserts in the above-mentioned detail data the summing field data obtained with reference to the above-mentioned layered structure information table based on summing field data and its delimiter was prepared and the above-

mentioned delimiter can be sorted as a sort key, the summary value about the summing field data of the upper layer can be obtained more at a high speed.
[0046] Moreover, since the step which chooses from summing field data the summing field data made applicable to a total was prepared, the summary value about the summing field data of the upper layer or a lower layer can be obtained more at a high speed from summing field data.
[0047] Since it constituted so that the order relation between the level name for every hierarchy, the element names belonging to the level, and those elements might be included in a layered structure information table further again, while being able to obtain the summary value about the summing field data of the upper layer or a lower layer from summing field data, the level of a total can be set up by the order relation between elements.
[0048] Moreover, since it constituted so that it might total for every combination of all containing the above-mentioned summing field data according detail data to selection of arbitration, it can respond to all retrieval conditions.
[0049] Furthermore, since it constituted so that zero data might be held as a summary value when a summary value was zero, it can grasp easily that a summary value is zero.

[Translation done.]

TECHNICAL FIELD

[Field of the Invention] Especially this invention relates to amelioration of the data-processing approach of being suitable for query processing of a multi dimensional database managerial system, and total processing, about the data-processing approach.

[Translation done.]

PRIOR ART

[Description of the Prior Art] Generally, in order to pull out multi-dimension data at a high speed, an index is used for what is called a multi dimensional database, and it is performing data processing. There are the following two kinds also in the approach using an index. The multi dimensional database which generates internally the data which unlike the relational database holding the data of description level load beforehand the data of minimum total level required of multi-dimension actuation from a relational database etc., and are used by multi-dimension actuation based on these data. To the data which such a multi dimensional database holds, an index is generated simultaneously at the time of loading of data, or an internal generate time, and the inquiry of multi-dimension actuation is performed at a high speed.
[0003] Moreover, the multi dimensional database which holds the data of detail level like the usual relational database, extends relational database actuation, and performs multi-dimension actuation. In such a multi dimensional database, the index needed by the join used by multi-dimension actuation is generated beforehand, and the inquiry of multi-dimension actuation is performed at a high speed. There are two kinds of approaches described above.
[0004] In order to pull out the conventional multi-dimension data which were described above at a high speed, there is an approach indicated by JP,4-229371,A as an example which uses an index and is performing data processing. Although changing and searching the hierarchy level of provisions of classification is indicated, the provisions-of-classification list which is equivalent to an index as a retrieval means is required of this example.

[Translation done.]

EFFECT OF THE INVENTION

[Effect of the Invention] Since this invention is constituted as explained above, it does effectiveness as taken below so.
[0043] Since it has the layered structure information table which set up the layered structure between summing field data beforehand, the summary value about the summing field data of the upper layer or a lower layer can be obtained from summing field data.
[0044] Moreover, since the step which totals detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data was prepared, the summary value about the summing field data of the upper layer can be obtained more at a high speed.
[0045] Furthermore, since the step which inserts in the above-mentioned detail data the summing field data obtained with reference to the above-mentioned layered structure information table based on summing field data and its delimiter was prepared and the above-mentioned delimiter can be sorted as a sort key, the summary value about the summing field data of the upper layer can be obtained more at a high speed.
[0046] Moreover, since the step which chooses from summing field data the summing field data made applicable to a total was prepared, the summary value about the summing field data of the upper layer or a lower layer can be obtained more at a high speed from summing field data.
[0047] Since it constituted so that the order relation between the level name for every hierarchy, the element names belonging to the level, and those elements might be included in a layered structure information table further again, while being able to obtain the summary value about the summing field data of the upper layer or a lower layer from summing field data, the level of a total can be set up by the order relation between elements.
[0048] Moreover, since it constituted so that it might total for every combination of all containing the above-mentioned summing field data according detail data to selection of arbitration, it can respond to all retrieval conditions.
[0049] Furthermore, since it constituted so that zero data might be held as a summary value when a summary value was zero, it can grasp easily that a summary value is zero.

TECHNICAL PROBLEM

[Problem(s) to be Solved by the Invention] Since the conventional data-processing approach is constituted as mentioned above, to perform multi-dimension actuation from the data of detail level, it is necessary to generate an index as pretreatment. In the case of the former which loads beforehand the data of minimum total level still more nearly required of multi-dimension actuation from a relational database etc., it is necessary to load the data of total level from the data of detail level. When detail data new every day are added to a large quantity, these pretreatments must pretreat each time and have the problem that become very heavy processing, become weekly and monthly processing actually, and the sex is lost instancy, in a mass multi dimensional database.
[0006] Also when it was made in order to cancel the above troubles, and performing multi-dimension actuation to the data of detail level, this invention makes pretreatment of generating an index unnecessary, and aims at offering a multi-dimension actuation function easily.

[Translation done.]

MEANS

[Means for Solving the Problem] The data-processing approach concerning this invention is equipped with the data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. By using the step which searches the summing field data of the upper layer or a lower layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, and the obtained summing field data as a key, the above-mentioned detail data are sorted and total processing is carried out at the above-mentioned summing field data unit.

[0008] Moreover, it has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which totals the above-mentioned detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data, By using the step which searches the summing field data of the upper layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, and the obtained summing field data as a key, the above-mentioned total data are sorted and total processing is carried out at the above-mentioned summing field data unit.

[0009] Furthermore, it has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which totals the above-mentioned detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data, The step which searches the summing field data of the upper layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, By using the step and the above-mentioned delimiter which insert the obtained summing field data and its delimiter in the above-mentioned detail data as a key, the above-mentioned total data are sorted and total processing is carried out at the above-mentioned summing field data unit.

[0010] Moreover, it has a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which totals, and the layered structure information table which set up beforehand the layered structure between the above-mentioned summing field data. The step which totals the above-mentioned detail data for every possible combination of all containing all the above-mentioned summing field data, and creates total data, The step which chooses from the above-mentioned summing field data the summing field data made applicable to a total, The step which searches the summing field data of the upper layer or a lower layer from the summing field data with reference to the above-mentioned layered structure information table based on the above-mentioned summing field data, By using the step and the above-mentioned delimiter which insert the obtained summing field data and its delimiter in the above-mentioned detail data as a key, the above-mentioned total data are sorted and

total processing is carried out at the above-mentioned summing field data unit.
[0011] The order relation between the level name for every hierarchy, the element names belonging to the level, and those elements is included in the above-mentioned layered structure information table further again.
[0012] Moreover, it totals for every combination of all containing the above-mentioned summing field data according the above-mentioned detail data contained in the element arranged by the sequence set up beforehand to selection of arbitration.
[0013] Furthermore, when the summary value by the above-mentioned total processing is zero, zero data are held as a summary value.
[0014]
[Embodiment of the Invention]
The gestalt 1 of operation of this invention is explained to a detail based on a drawing below gestalt 1. of operation. Completion of install and a setup of the detail database (Detail DB is called below) system in the gestalt 1 of this operation generates one environment shown in drawing 1 . In drawing 1 , 1 is a catalog and it is one set of PC. Only one environment can be generated in Server and a catalog 1 exists in the environment one **. Under a catalog 1, it is possible to generate and delete two or more schemas 2. All the information about the schema 2 grade generated under a catalog 1 is managed by the above-mentioned detail DB system as catalog 1 information.
[0015] To a schema 2, it is possible to generate and delete two or more real tables 3. In a schema 2, it is SET. A default schema can be set up by SCHEMA. When not setting up an explicit default schema, the schema decided by the system turns into a default schema.
[0016] To each schema 2, it is possible to generate and delete the category 4 as a layered structure information table. In a category 4, the sequence between the elements (actual-key value included in the real table 3) in data-hierarchy structure and each hierarchy is expressed, and it consists of level of N individual to which the identifier was attached. It has a list of the element belonging to the level, and the order relation between elements in each level. Moreover, between level, it has the relation for every element and this expresses hierarchical relationship. Whether a train is seen on the level of which category 4 throat sets up in the category function on the original database language in a detail DB system. Drawing 2 is an example which shows the structure of a category 4, and 11 shows "level", such as a section and a name, and the "sequence" of each element [in / 12 / "elements", such as the Administrative Division and the personnel division, and 13 and / in 14 / the same level]. [the "relation" between each element]
[0017] The real table 3 is a sequential file in the fixed length who consists of two or more trains 5 (field). The table which can be defined as a detail DB system is only the real table 3. However, the "category function" which realizes the imagination train 5 which can change the view from the program of each train free is set up.
[0018] By applying a "category function" to a train 5, it becomes possible to access a real table in the imagination train 5. Only by defining the view (view) of a train 5 through a category 4, the treatment of each train 5 can be changed for every application program. It is made to look as if the train 5 which is not held directly existed on the real table 3 by specifically applying the above-mentioned category 4 to the train 5 which constitutes the real table 3 using a category function. Thereby, modification of the list of a key, selection of total level, etc. are attained.
[0019] Next, it explains that processing of the whole multi-dimensional processing flows. In many cases, what is called a dimension in the world of a multi dimensional database holds hierarchical structure, and various retrieval and totals on hierarchy level are possible for it. For example, if a personal name is made into an example, when the individual belongs to the firm, if an identifier is decided, it will be decided that the section and the section to which the man belongs will be a meaning. That is, if the layered structure of this personal name is independently defined even when only the personal name is held in the train 5 of the real table 3, the retrieval and total processing which used as the key the section which is a hierarchy's upper level, and a section will be attained. In the detail DB system in the gestalt 1 of this operation, these actuation can be performed to the real table 3 of the detail base. In order to realize the same function as the

dimension in such a multi dimensional database especially, the view of the real table 3 which is raw data is changeable free by introducing the category 4 which is the layered structure of an imagination train.
[0020] Drawing 3 shows the outline of multi-dimensional processing, 3 shall be a real table and only the summing field of a "manufacturer", a "date", and "sales" should exist at the beginning. Next, the demand made applicable to a total also about a "branch office" and a "product" occurs, and based on the category 4 of a hierarchy 22, the imagination train 23 is processed as if it existed in the real table 3. Next, the multi-dimension total result 24 of having totaled about all the summing fields of the real table 3 is obtained by totaling after a sort as a sort key about each summing field of the real table 3, respectively. All the above actuation is controlled by the application program 25.
[0021] The train 5 which constitutes the real table 3 is roughly divided, and is classified into two. One is a train holding the information for identifying an identifier of an object which is expressed with a master code. As an example, although a personnel code, an identifier, etc. are mentioned, these are used, in order that both may identify a personal name. The second is a train holding the information for identifying a condition of an object which is expressed numerically. A selling price, the sale number, etc. are mentioned as this example. Although these former was called the "dimension", the latter was called the "variable" and treatment is clearly divided in the general multi dimensional database, both these are intermingled in one line in the common world and the common detail DB system of SQL. In addition, it is GROUP when performing total processing by GROUPBY in a detail DB system. The train which the train specified as the key of BY specifies as a dimension and a set function corresponds to a variable.
[0022] Next, detail explanation is given about a category 4. It becomes possible to acquire the same effectiveness as the "dimension" in the many dimensions DB by applying a category 4 to the train 5 with the real table 3. The following information is included in the category.
(1) In case the concrete value category name of the special values ("ALL" etc.) of related list (4) each level of the element between the element list belonging to the level name for every category name (2) hierarchy and its level and the order relation (3) level between elements and a level name apply a category function, use them, the category name is unique within a schema, and if the level name is unique within a category, it is good. The element 12 of each level is an item used as the key in retrieval or a total, and can define sequence between the elements of the same level. However, this sequence is the case where have semantics only when ROLLUP and a CUBE total are performed, and it does not total, and simple GROUP. In BY, the sequence for every usual data type is applied. Thereby, setting out becomes possible for every user about the order of an output as a result of the below-mentioned roll-up total.
[0023] The layered structure as a category 4 is defined by defining relation with the element 12 of the level of one high order at the time of the element definition of each level. Retrieval and a total with an upper level are attained by defining this relation. It is also possible to prepare the category for oneself and to define only the required element 12 as an element 12 of level positively, when what he should process in the element 12 of a train 5 as an approach using this function is limited.
[0024] Although the special value "ALL" exists logically in a category 4, this is used as a value showing the "intermediate total" and the "major total" at the time of performing a roll-up and a cube total. Drawing 4 is what showed matching with the real table 3 and a category 4, and since Taro Yamada in the real table 3 exists in a category 4, he can match as it is.
[0025] In the detail DB system in the gestalt 1 of this operation, the category function which realizes the imagination train which does not exist actually is offered to the usual train which constitutes the real table 3. In order to use a category function, after defining the above-mentioned category 4, it is required to make the level of an either a train 5 or the category 4 map to the real table 3 which applies it. Since it is possible to define two or more layered structures by changing a view to one train 5, the relation between a category 4 and a train 5 is set to N to 1. This shows that two or more completely different categories are applicable to <DP N=0005> like an "organization" and an "executive" to the train of an identifier. Moreover, it is possible to define two or more level in one category 4, and a category function can be applied to

each of that level. This can define two or more level called the "section", a "section", and "name" in a category called an organization, and shows that a category function is applicable on every level in it.
[0026] The sequence 14 between the elements 12 defined within the category 4 is effective only when ROLLUP and a CUBE total are performed to the train acquired with the application of the category function. Moreover, it not only maps a value to the element 12 of each level in a category 4, but a category function can be mapped to the subelement of each level.
[0027] Drawing 5 is what showed relation with the imagination train acquired by applying a train and a category, and by applying the category function to the organization (section) from an identifier shows notionally that the imagination train of a section name is generated and the imagination train of a section name is generated by applying the category function to the organization (section) from an identifier.
[0028] Drawing 6 is what showed the relation between a train and an imagination train concretely, and shows signs that the identifier on a train is generated as a section name on an imagination train by applying the category function to the organization (section) from an identifier.
[0029] The data type offered by the detail DB system in the gestalt 1 of this operation offers a "DATE mold" with the function similar to the view of a category. This DATE mold is the data type showing the date. Drawing 7 shows the layered structure of a DATE mold, and shows the layered structure by the tacit relation by the established practice from the former.
[0030] As the approach of gestalt 2. many dimensions total processing operation, they are ROLLUP and CUBE. Although mentioned, the gestalt 2 of the operation which applied this invention is explained to these based on drawing. GROUP in the database language of a detail DB system the total processing which used two or more specified keys as the 1 lump when two or more keys were specified and grouping was performed in BY — only carrying out (subtotal) — it is . If either of two or more specified keys takes the 1 surroundings (key breaking) by the total of ROLLUP or CUBE to it, total processing to there will be totaled further (intermediate total and major total), and it will return as a result. The difference in the output of ROLLUP and CUBE is shown in drawing 8.. "*" in drawing means all the elements contained in a train, and "ALL" means the sum total of all the elements of the train. The example of an expression of drawing 8 shows a thing without hierarchical relationship according to a three dimension (the key of a GROUP BY phrase is three pieces).
[0031] In order to consider the concrete example of ROLLUP and CUBE, the real table 3 with the category 4 shown in drawing 9 is considered. Moreover, the structure of the real table 3 is constituted as a data file by sequential organization of the detail data which consist of two or more summing field data used as one unit which shows drawing 10 and totals. Moreover, SQL is used as a database language.
[0032] GROUP shown in drawing 11 in order to obtain "the monthly field option raising table from January of the Katase branch office to March" to the real table 3 shown in drawing 10 The result obtained when the sentence which specifies retrieval of three kinds of detail databases, BY, ROLLUP, and CUBE, is published is the chart shown in drawing 12 . In drawing 12 (a) GROUP The total by BY, a total according [(b)] to ROLLUP, and (c) show the total result by CUBE, and television, video, a washer, a refrigerator, AV, household electric appliances, and the point that each totalizer of the sales in January - March is totaled are the descriptions in 1Q column in (c). In addition, "CATEGORY-" shown in each SQL sentence of drawing 11 shows a category function, and relating with the upper layer by the category 4 is performed.
[0033] Here, the art for asking for the roll-up and cube which are many dimensions-actuation is explained. The approach of asking for a roll-up focusing on sorting application is GROUP of SQL focusing on sorting application fundamentally. After resembling the processing which performs BY and sorting input data, a roll-up total can be obtained by performing total processing for every key breaking. In order to ask for the cube of N dimension focusing on sorting application, roll-up total processing can be obtained by repeating N times.
[0034] First, if direct roll-up total processing is performed to detail data, since a total overhead may become large, depending on the number of cases of detail data, it is the usual GROUP as a

first stage story. It will ask for BY total. In drawing 12 , it has totaled by (a) for every possible combination of all containing television, video, a washer, a refrigerator, AV, household electric appliances, and all the summing field data of the moon. This GROUP It is totaled by a field and every month as it is shown in (b), when the roll-up total to a field and the moon is performed as a second stage story to the data (or detail data) which carried out BY. In such roll-up actuation, when same actuation is performed for every dimension, all cube total results will be obtained in a culmination.

[0035] Here, when a summary value is zero, zero data are held as a summary value by making a summary value into zero. In drawing 13 , when a **** branch office and the Imaizumi branch office are sale zero, the zero total which adds the data of sale zero and totals can be carried out. This zero summarize function can be carried out by assignment, and can choose ****.

[0036] Gestalt 3. this invention of operation is explained as a gestalt 3 of operation about the case where it applies to a pipeline merge sorter. Drawing 14 shows a part for the body of the data flow when performing the total by CUBE in a pipeline merge sorter. Setting to drawing, 31 is GROUP. It is what chose from data after performing the total by BY the element made applicable to a total, and stored those summing field data in the disk buffer (not shown), and this data reads the data of two or more affairs into a pipeline merge sorter (not shown) inner buffer at once by the DMA transfer. Projection processing is performed in a pipeline merge sorter, and it will be in the condition which the content of the record is extended to sort key data (left-hand side in a buffer), and the data (right-hand side in a buffer) returned to a user, and shows in 32. Next, the sequence key for a sort and high order hierarchy data as a delimiter will be in the condition of the buffer which it is written in in a buffer and shown in 33 with the firmware (F/W) in a pipeline merge sorter, making a category 4 reference in the clearance left with the projection at the time of a DMA transfer. If all buffers are buried, data will be sent to the sort section 34 and will be sorted in order of the sort key on the left-hand side of a buffer. When lower layer data are written in in a buffer from a category 4, the content of processing is the same only by the location written in turning into a different location from the condition which shows in 33.

[0037] The result sorted in order of the sort key on the left-hand side of a buffer will be in the condition of the buffer shown in 35. Although total processing is performed by F/W, the total (roll-up total) by key breaking is performed here again. A total in a product and a field is completed by finally returning the content of the record of the right-hand side in a buffer to a user. Cube processing is completed by performing about all the elements that chose the above processing as the date, a branch office, and a manufacturer's object for a total.

[0038] Next, actuation of the gestalt 3 of operation is explained. Drawing 15 is GROUP which is a processing flow explaining actuation, totals for every possible combination of all that contains all summing field data from the detail data of the real table 3 first, and creates total data. BY is performed (step S1). Next, the summing field data made applicable to a total are chosen from summing field data, and it stores in a disk buffer (step S2). Since the summing field data which were not chosen are deleted, the record length will be in the condition of having been reduced. Here, there are few elements, it is possible also for storing all summing field data in a disk buffer, without choosing, does not need to search lower layer data from a category 4 in this case, and serves as retrieval of only upper layer data.

[0039] Next, the tooth space which extends the content of the record to the data returned to sort key data and a user by projection processing, and inserts the summing field data and the sequence key of the upper layer of a category or a lower layer is set up (step S3). A category is searched and the summing field data and its sequence key of the upper layer or a lower layer are inserted in the above-mentioned tooth space (step S4). A sequence key is sorted as a sort key (step S5).

[0040] Next, it totals about the summing field data made into the object of a sort key by the set operation, and the line for a total is added (step S6). Here, it judges whether total processing was ended about all the summing field data of the selected element (step S7), and when having not ended, the sequence key of the summing field data which carry out total processing is stored in step S3 at a disk buffer at return and a degree. When it ends, CUBE about the summing field data moved and chosen as step S8 is completed.

[0041] Sorting application is more possible at a high speed by not using the comparatively long alphabetic data of a data length as a sort key, but sorting the sequence key which is a code in a figure as a sort key as above-mentioned.

[Translation done.]

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]
[Drawing 1] It is detail DB structure-of-a-system drawing showing the gestalt 1 of implementation of this invention.
[Drawing 2] It is structural drawing of a category showing the gestalt 1 of implementation of this invention.
[Drawing 3] It is the outline explanatory view of multi-dimensional processing showing the gestalt 1 of implementation of this invention.
[Drawing 4] It is the explanatory view showing the relation of the category and real table showing the gestalt 1 of implementation of this invention.
[Drawing 5] It is the block diagram of the train which shows the gestalt 1 of implementation of this invention, and an imagination train.
[Drawing 6] It is an explanatory view explaining the relation between the train which shows the gestalt 1 of implementation of this invention, and an imagination train.
[Drawing 7] It is structural drawing showing the layered structure of the DATE mold in which the gestalt 1 of implementation of this invention is shown.
[Drawing 8] It is comparison drawing showing a difference of the total approach which shows the gestalt 2 of implementation of this invention.
[Drawing 9] It is the block diagram showing the content of the category which shows the gestalt 2 of implementation of this invention.
[Drawing 10] It is structural drawing of a detail data file showing the gestalt 2 of implementation of this invention.
[Drawing 11] It is the list of SQL sentences in which the gestalt 2 of implementation of this invention is shown.
[Drawing 12] It is the output list which shows three kinds of total results which show the gestalt 2 of implementation of this invention.
[Drawing 13] It is the explanatory view of a zero total showing the gestalt 2 of implementation of this invention.
[Drawing 14] It is the data flow diagram showing the gestalt 3 of implementation of this invention.

[Drawing 15] It is the flow chart which shows the gestalt 3 of implementation of this invention.
[Description of Notations]
3 A real table (detail data), 4 A category (layered structure information table), 12 Element.

[Translation done.]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（A）　(11)特許出願公開番号

特開平10－301935

(43)公開日　平成10年(1998)11月13日

(51)Int.Cl.[6]	識別記号	ＦＩ	
Ｇ０６Ｆ　17/30		Ｇ０６Ｆ　15/403	３６０Ｚ

審査請求　有　　請求項の数７　ＯＬ　（全 11 頁）

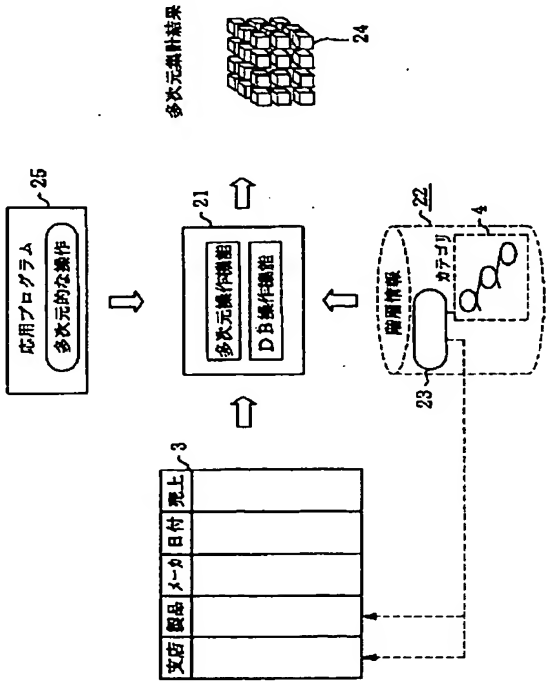
(21)出願番号　特願平9－104830

(22)出願日　平成９年(1997)４月22日

(71)出願人　000006013
三菱電機株式会社
東京都千代田区丸の内二丁目２番３号
(72)発明者　岩▲崎▼　浩文
東京都千代田区丸の内二丁目２番３号　三菱電機株式会社内
(72)発明者　安藤　隆朗
東京都千代田区丸の内二丁目２番３号　三菱電機株式会社内
(74)代理人　弁理士　宮田　金雄　（外２名）

(54)【発明の名称】　データ処理方法

(57)【要約】
【課題】　明細レベルのデータに対し多次元操作を行う場合にも、索引を生成する等の前処理を不要とし、容易に多次元操作を行うデータ処理方法を提供する。
【解決手段】　階層情報22のカテゴリ４に基づき、実表３に存在するデータに含まれる「支店」「製品」に関連するデータを得て仮想的な列23として実表３に追加する。次に、実表３の各集計項目についてそれぞれソートキーとしてソート後集計することにより、実表３の全ての集計項目について集計した多次元集計結果24を得る。



【特許請求の範囲】
【請求項1】　集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層又は下位層の集計項目データを検索するステップ、得られた集計項目データをキーとして上記明細データをソートし上記集計項目データ単位に集計処理することを特徴とするデータ処理方法。
【請求項2】　集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記明細データを全ての上記集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するステップ、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層の集計項目データを検索するステップ、得られた集計項目データをキーとして上記集計データをソートし上記集計項目データ単位に集計処理することを特徴とするデータ処理方法。
【請求項3】　集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記明細データを全ての上記集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するステップ、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層の集計項目データを検索するステップ、得られた集計項目データとその識別記号を上記明細データに挿入するステップ、上記識別記号をキーとして上記集計データをソートし上記集計項目データ単位に集計処理することを特徴とするデータ処理方法。
【請求項4】　集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記明細データを全ての上記集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するステップ、上記集計項目データから集計対象とする集計項目データを選択するステップ、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層又は下位層の集計項目データを検索するステップ、得られた集計項目データとその識別記号を上記明細データに挿入するステップ、上記識別記号をキーとして上記集計データをソートし上記集計項目データ単位に集計処理することを特徴とするデータ処理方法。
【請求項5】　上記階層構造情報テーブルには、階層ごとのレベル名、そのレベルに属する要素名、それらの要素間の順序関係を含むことを特徴とする請求項1～請求項4のいずれかに記載のデータ処理方法。
【請求項6】　予め設定された順序により配設された要素に含まれる上記明細データを任意の選択による上記集計項目データを含む全ての組み合わせごとに集計することを特徴とする請求項1～請求項5のいずれかに記載のデータ処理方法。
【請求項7】　上記集計処理による集計値がゼロのときはゼロデータを集計値として保持することを特徴とする請求項1～請求項6のいずれかに記載のデータ処理方法。
【発明の詳細な説明】
【0001】
【発明の属する技術分野】この発明は、データ処理方法に関し、特に多次元データベース管理システムの問合せ処理ならびに集計処理に適するデータ処理方法の改良に関するものである。
【0002】
【従来の技術】一般に、多次元データベースと呼ばれるものは、多次元データを高速に引き出すために、索引を用いてデータ処理を行っている。索引を用いる方法にも以下の2通りがある。明細書レベルのデータを保持するリレーショナルデータベースとは異なり、多次元操作で必要な最低限の集計レベルのデータを予めリレーショナルデータベース等からロードし、これらのデータを元に多次元操作で使用するデータを内部的に生成する多次元データベース。このような多次元データベースが保持するデータには、データのロード時または内部的な生成時に索引を同時に生成し、多次元操作の問合せを高速に行う。
【0003】また、通常のリレーショナルデータベースと同様に明細レベルのデータを保持しており、リレーショナルデータベース操作を拡張して多次元操作を行う多次元データベース。このような多次元データベースでは、多次元操作で使用されるジョインで必要とする索引を予め生成しておき、多次元操作の問合せを高速に行う。以上述べた2通りの方法がある。
【0004】以上述べた様な従来の多次元データを高速に引き出すために、索引を用いてデータ処理を行っている一例として、特開平4－229371号公報に開示された方法がある。この例では、分類項目の階層レベルを変えて検索することが開示されているが、検索手段として索引に相当する分類項目一覧が必要である。
【0005】
【発明が解決しようとする課題】従来のデータ処理方法は以上の様に構成されているので、明細レベルのデータから多次元操作を行う場合には、前処理として索引を生成する必要がある。さらに多次元操作で必要な最低限の集計レベルのデータを予めリレーショナルデータベース等からロードする前者の場合は、明細レベルのデータか

ら集計レベルのデータをロードする必要もある。これらの前処理は、日々新しい明細データが大量に追加される場合には、その都度前処理を実施せねばならず、大容量の多次元データベースでは非常に重たい処理となり、実際には週次、月次処理となるなどして即時性が失われているといった問題がある。

【0006】この発明は、上記のような問題点を解消するためになされたもので、明細レベルのデータに対し多次元操作を行う場合にも、索引を生成する等の前処理を不要とし、容易に多次元操作機能を提供することを目的とする。

【0007】

【課題を解決するための手段】この発明に係るデータ処理方法は集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層又は下位層の集計項目データを検索するステップ、得られた集計項目データをキーとして上記明細データをソートし上記集計項目データ単位に集計処理するものである。

【0008】また、集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記明細データを全ての上記集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するステップ、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層の集計項目データを検索するステップ、得られた集計項目データをキーとして上記集計データをソートし上記集計項目データ単位に集計処理するものである。

【0009】さらに、集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記明細データを全ての上記集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するステップ、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層の集計項目データを検索するステップ、得られた集計項目データとその識別記号を上記明細データに挿入するステップ、上記識別記号をキーとして上記集計データをソートし上記集計項目データ単位に集計処理するものである。

【0010】また、集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルと上記集計項目データ間の階層構造を予め設定した階層構造情報テーブルとを備え、上記明細データを全ての上記集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するステップ、上記集計項目データから集計対象とする集計項目データを選択するステップ、上記集計項目データに基づき上記階層構造情報テーブルを参照してその集計項目データより上位層又は下位層の集計項目データを検索するステップ、得られた集計項目データとその識別記号を上記明細データに挿入するステップ、上記識別記号をキーとして上記集計データをソートし上記集計項目データ単位に集計処理するものである。

【0011】さらにまた、上記階層構造情報テーブルには、階層ごとのレベル名、そのレベルに属する要素名、それらの要素間の順序関係を含むものである。

【0012】また、予め設定された順序により配設された要素に含まれる上記明細データを任意の選択による上記集計項目データを含む全ての組み合わせごとに集計するものである。

【0013】さらに、上記集計処理による集計値がゼロのときはゼロデータを集計値として保持するものである。

【0014】

【発明の実施の形態】

実施の形態1．以下、本発明の実施の形態1を図面に基づいて詳細に説明する。本実施の形態1における明細データベース（以下明細DBと称する）システムは、インストール及びセットアップを完了すると、図1に示す一つの環境が生成される。図1において、1はカタログであり、1台のPC Serverには、一つの環境だけが生成可能であり、その環境にはカタログ1が唯一つ存在する。カタログ1下には、複数のスキーマ2を生成、削除することが可能である。カタログ1下に生成されるスキーマ2等に関する全ての情報は、カタログ1情報として上記明細DBシステムにより管理される。

【0015】スキーマ2には、複数の実表3を生成、削除することが可能である。スキーマ2には、SET SCHEMAによりデフォルトスキーマを設定できる。明示的デフォルトスキーマを設定しない場合は、システムで決められたスキーマがデフォルトスキーマになる。

【0016】各スキーマ2には、階層構造情報テーブルとしてのカテゴリ4を生成、削除することが可能である。カテゴリ4とは、データの階層構造と各階層での要素（実表3に含まれる実キー値）間の順序を表現したものであり、名前の付いたN個のレベルから構成される。各レベルには、そのレベルに属する要素の一覧と要素間の順序関係を持つ。また、レベル間では、要素毎の関係を持ち、それにより階層関係を表現する。どのカテゴリ4のどのレベルで列を見るかは、明細DBシステムにおける独自のデータベース言語上のカテゴリ関数において設定する。図2はカテゴリ4の構造を示す一例で、11は課、名等の「レベル」、12は総務部、人事課等の「要素」、13は各要素間の「関係」、14は同一レベルにおけ

る各要素の「順序」を示している。

【００１７】実表３は、複数の列５（フィールド）から構成される、固定長で順次ファイルである。明細ＤＢシステムに定義できる表は実表３のみである。但し、各列のプログラムからの見方を自在に変更することができるような仮想的な列５を実現する「カテゴリ関数」を設定する。

【００１８】列５に対して「カテゴリ関数」を適用することにより、実表を仮想的な列５でアクセスすることが可能となる。列５の見方（ビュー）をカテゴリ４を通して定義するだけで、各列５の扱いを応用プログラム毎に変更することができる。具体的には、上記カテゴリ４をカテゴリ関数を用いて実表３を構成する列５に適用することにより、実表３上に直接保持しない列５が、あたかも存在するかのように見せかけるものである。これにより、キーの並びの変更や集計レベルの選択等が可能となる。

【００１９】次に、多次元処理全体の処理の流れについて説明する。多次元データベースの世界で次元と呼ばれるものは、多くの場合階層的な構造を保持し、種々の階層レベルでの検索や集計が可能である。例えば、個人名を例にすると、その個人が会社に属している場合は、名前が決まるとその人の属する課や部が一意に決まる。つまり、もし実表３の列５に個人名しか保持していない場合でも、この個人名の階層構造が別に定義されていれば、階層の上位レベルである部や課をキーとした検索や集計処理が可能となる。本実施の形態１における明細ＤＢシステムでは、これらの操作を明細ベースの実表３に対して実行することができる。特に、このような多次元データベースにおける次元と同様の機能を実現するために仮想的な列の階層構造であるカテゴリ４を導入することにより、生データである実表３の見方を自在に変えることができる。

【００２０】図３は多次元処理の概要を示すもので、３は実表であり、当初は「メーカ」「日付」「売上」の集計項目しか存在しなかったものとする。次に、「支店」「製品」についても集計対象とする要求が発生し、階層情報22のカテゴリ４に基づき、仮想的な列23が、あたかも実表３に存在するかのように処理される。次に、実表３の各集計項目についてそれぞれソートキーとしてソート後集計することにより、実表３の全ての集計項目について集計した多次元集計結果24を得る。以上の動作は、すべて応用プログラム25により制御される。

【００２１】実表３を構成する列５は、大きく分けて二つに分類される。一つは、マスタコードで表されるような物の名前を識別するための情報を保持する列である。例としては、社員コードや名前等が挙げられるが、これらは両方とも個人名を識別するために用いられる。二つめは、数値で表されるような物の状態を識別するための情報を保持する列である。この例としては、売り値や販売個数等が挙げられる。一般的な多次元データベースでは、これらの前者を「次元」、後者を「変数」と呼び、明確に扱いを分けているが、一般的なＳＱＬの世界及び明細ＤＢシステムでは、これら両方が一つの行に混在する。なお、明細ＤＢシステムにおいてGROUPBYで集計処理を行う場合、GROUP　BYのキーに指定する列が次元、集合関数に指定する列が変数に該当する。

【００２２】次に、カテゴリ４について詳細説明する。実表３のある列５にカテゴリ４を適用することで、多次元ＤＢにおける「次元」と同様の効果を得ることが可能となる。カテゴリには、以下の情報が含まれている。

（１）カテゴリ名

（２）階層毎のレベル名、及びそのレベルに属する要素一覧、及び要素間の順序関係

（３）レベル間の要素の関係一覧

（４）各レベルの特殊値（「ＡＬＬ」等）の具体値

カテゴリ名及びレベル名は、カテゴリ関数を適用する際に用い、カテゴリ名はスキーマ内でユニークであり、レベル名はカテゴリ内でユニークであれば良い。各レベルの要素12は、検索や集計におけるキーとなる項目であり、同一レベルの要素間では順序を定義できる。但し、この順序はROLLUP及びCUBE集計を行った場合のみ意味を持ち、集計を行わない場合や単純なGROUP　BYの場合には、通常のデータ型毎の順序が適用される。これにより、後述のロールアップ集計の結果出力順を利用者毎に設定可能となる。

【００２３】各レベルの要素定義時に一つ上位のレベルの要素12との関係を定義することにより、カテゴリ４としての階層構造を定義する。この関係を定義することにより、上位レベルでの検索や集計が可能となる。この機能を積極的に使う方法として、列５の要素12の中で自分が処理すべきものが限定される場合には、自分用のカテゴリを用意し、必要な要素12のみをレベルの要素12として定義することも可能である。

【００２４】カテゴリ４内には、「ＡＬＬ」という特殊値が論理的に存在するが、これはロールアップやキューブ集計を実行した場合の「中計」や「大計」を表す値として用いられる。図４は、実表３とカテゴリ４との対応付けを示したもので、実表３中の山田太郎はカテゴリ４中に存在するので、そのまま対応付け可能である。

【００２５】本実施の形態１における明細ＤＢシステムでは、実表３を構成する通常の列に対し、実際には存在しない仮想的な列を実現するカテゴリ関数を提供する。カテゴリ関数を利用するためには、前述のカテゴリ４を定義した後に、それを適用する実表３に対して、列５とカテゴリ４内のいずれかのレベルをマッピングさせることが必要である。一つの列５に対して見方を変えることにより、階層構造を複数定義することが可能であるため、カテゴリ４と列５の関係はN対１になる。これは、名前という列に対して、「組織」と「役職」というよう

に全く異なる複数のカテゴリを適用できることを示す。また一つのカテゴリ４内には複数のレベルを定義することが可能であり、その各レベルに対してカテゴリ関数を適用できる。これは、組織というカテゴリの中には、「部」と「課」と「名」という複数のレベルを定義でき、その中のどのレベルでもカテゴリ関数を適用できることを示す。

【００２６】カテゴリ４内で定義した要素12間の順序14は、カテゴリ関数を適用して得られた列に対してROLLUPやCUBE集計を行った場合のみ有効である。また、カテゴリ関数はカテゴリ４内の各レベルの要素12に値をマッピングするだけでなく、各レベルの副要素にもマッピングすることが可能である。

【００２７】図５は列とカテゴリを適用することにより得られる仮想的な列との関係を示したもので、名前から組織（課）へのカテゴリ関数を適用することにより課名の仮想的な列が生成され、名前から組織（部）へのカテゴリ関数を適用することにより部名の仮想的な列が生成されることを概念的に示したものである。

【００２８】図６は列と仮想的な列との関係を具体的に示したもので、列上の名前が、名前から組織（課）へのカテゴリ関数を適用することにより仮想的な列上に課名として生成される様子を示している。

【００２９】本実施の形態１における明細ＤＢシステムで提供するデータ型は、カテゴリの考え方に似た機能を持つ「DATE型」も提供する。このDATE型は、日付を表すデータ型である。図７はDATE型の階層構造を示すもので、従来からの慣習による暗黙の関係による階層構造を示している。

【００３０】実施の形態２．多次元的な集計処理の方法として、ROLLUPとCUBE　が挙げられるが、これらに本発明を適用した実施の形態２について図に基づいて説明する。明細ＤＢシステムのデータベース言語の中のGROUP　BYにおいて、複数のキーを指定してグループ化を行った場合、指定した複数のキーを一固まりとした集計処理を行う（小計）だけである。それに対してROLLUPやCUBEの集計では、指定した複数のキーのいずれかが一周り（キーブレーク）すると、そこまでの集計処理をさらに集計（中計や大計）し、結果として返すものである。ROLLUPとCUBEの出力結果の違いを図８に示す。図中の「＊」は、列に含まれる全ての要素を意味し、「ＡＬＬ」は、その列の要素全ての合計を意味する。図８の表現例は、３次元（GROUP　BY句のキーが３個）で階層関係がないものを示す。

【００３１】ROLLUPとCUBEの具体的な例を考えるため、図９に示すカテゴリ４を持つ実表３を考える。また、実表３の構造は図１０に示す様になっており、集計する一単位となる複数の集計項目データから構成される明細データの順編成によるデータファイルとして構成されている。また、データベース言語としてはSQLを使用している。

【００３２】図１０に示す実表３に対して、「片瀬支店の１月から３月までの月別分野別売り上げ表」を得るために、図１１に示すGROUP　BY、ROLLUP、CUBEの３通りの明細データベースの検索を指定する文を発行した場合に得られる結果が図１２に示す一覧表である。図１２において、（ａ）が　GROUP　BYによる集計、（ｂ）がROLLUPによる集計、（ｃ）がCUBEによる集計結果を示しており、（ｃ）においては、１Ｑ欄において、テレビ、ビデオ、洗濯機、冷蔵庫、ＡＶ、家電、１月～３月の売上のそれぞれの総合計が集計されている点が特徴である。なお、図１１の各SQL文に示されている「ＣＡＴＥＧＯＲＹ～」がカテゴリ関数を示し、カテゴリ４による上位層への関係付けが行われる。

【００３３】ここで、多次元的な操作であるロールアップやキューブを求めるための処理方法について説明する。ソート処理を中心にロールアップを求める方法は、基本的にはソート処理を中心にＳＱＬのGROUP　BYを行う処理に似ており、入力データをソートした後キーブレーク毎に集計処理を行うことでロールアップ集計を得られる。ソート処理を中心にＮ次元のキューブを求めるためには、ロールアップ的な集計処理をＮ回繰り返すことで得ることができる。

【００３４】まず、明細データに対して直接ロールアップ的な集計処理を実行すると、集計オーバヘッドが大きくなる可能性があるため、明細データの件数によっては、第一段階として通常のGROUP　BY集計を求めることになる。図１２において、（ａ）では、テレビ、ビデオ、洗濯機、冷蔵庫、ＡＶ、家電、月の全ての集計項目データを含む可能な全ての組み合わせごとに集計してある。このGROUP　BYしたデータ（または、明細データ）に対して、第二段階として分野、月に対するロールアップ集計を行うと、（ｂ）に示す通り、分野、月ごとに集計される。このようなロールアップ操作を各次元毎に同様の操作を行うと、最終段階では、全てのキューブ集計結果が得られることになる。

【００３５】ここで、集計値がゼロの場合は集計値をゼロとしてゼロデータを集計値として保持する。図１３において、辻堂支所と今泉支所が売上高ゼロの場合、売上高ゼロのデータを追加して集計するゼロ集計をすることができる。このゼロ集計機能は、指定により実施するしないを選択することができる。

【００３６】実施の形態３．本発明をパイプライン・マージソーターに適用した場合について実施の形態３として説明する。図１４はパイプライン・マージソーターにおいて、CUBEによる集計を行うときのデータフローの主要部分を示したものである。図において、31はGROUP　BYによる集計を行った後のデータから集計対象とする要素を選択しそれらの集計項目データをディスクバッファ（図示せず）に格納したもので、このデータはＤＭＡ転

送によりパイプライン・マージソーター（図示せず）内バッファに複数件のデータを一度に読み込む。パイプライン・マージソーター内において射影処理を実行し、ソートキーデータ（バッファ内左側）とユーザに返すデータ（バッファ内右側）にレコードの内容を拡張し32に示す状態となる。次に、パイプライン・マージソーター内のファームウェア（Ｆ／Ｗ）により、DMA転送時の射影で空けられた隙間に、カテゴリ４を参照にしながら識別記号としてのソート用順序キーや上位階層データがバッファ内に書き込まれ33に示すバッファの状態になる。バッファが全て埋まると、データはソート部34へ送られ、バッファ左側のソートキー順にソートされる。カテゴリ４から下位層データがバッファ内に書き込まれる場合には、書き込まれる位置が33に示す状態とは異なる位置になるだけで処理内容は同一である。

【００３７】バッファ左側のソートキー順にソートされた結果は35に示すバッファの状態になる。ここでまたＦ／Ｗにより、集計処理が行われるが、ここではキーブレークでの集計（ロールアップ集計）が行われる。最後にバッファ内右側のレコードの内容をユーザに返すことにより、製品、分野での集計が完了する。以上の処理を日付、支店、メーカの集計対象として選択した全ての要素について実行することによりキューブ処理が完了する。

【００３８】次に、実施の形態３の動作について説明する。図１５は、動作を説明する処理フローであり、先ず、実表３の明細データから全ての集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するGROUP　BYを実行する（ステップＳ１）。次に、集計項目データから集計対象とする集計項目データを選択し、ディスクバッファに格納する（ステップＳ２）。選択されなかった集計項目データは削除されるので、レコード長は縮小された状態となる。ここで、要素数が少なく、選択せずに全ての集計項目データをディスクバッファに格納することも可能であり、この場合にはカテゴリ４から下位層データを検索する必要はなく、上位層データのみの検索となる。

【００３９】次に、射影処理によりソートキーデータとユーザに返すデータにレコードの内容を拡張しカテゴリの上位層又は下位層の集計項目データ及び順序キーを挿入するスペースを設定する（ステップＳ３）。カテゴリを検索して上位層又は下位層の集計項目データとその順序キーを上記スペースに挿入する（ステップＳ４）。順序キーをソートキーとしてソートする（ステップＳ５）。

【００４０】次に、集合演算によりソートキーの対象とした集計項目データについて集計し集計用の行を追加する（ステップＳ６）。ここで、選択した要素の集計項目データの全てについて集計処理を終了したかを判定し（ステップＳ７）、終了していない場合はステップＳ３に戻り、次に集計処理する集計項目データの順序キーをディスクバッファに格納する。終了した場合はステップＳ８に移り、選択した集計項目データについてのCUBEが完成する。

【００４１】上記の通り、データ長の比較的長い文字データをソートキーとせず、数字によるコードである順序キーをソートキーとしてソートすることにより、より高速にソート処理が可能である。

【００４２】

【発明の効果】この発明は、以上説明したように構成されているので、以下に示すような効果を奏する。

【００４３】集計項目データ間の階層構造を予め設定した階層構造情報テーブルを備えているので、集計項目データより上位層又は下位層の集計項目データについての集計値を得ることができる。

【００４４】また、明細データを全ての上記集計項目データを含む可能な全ての組み合わせごとに集計し集計データを作成するステップを設けたので、より高速に上位層の集計項目データについての集計値を得ることができる。

【００４５】さらに、集計項目データに基づき上記階層構造情報テーブルを参照して得られた集計項目データとその識別記号を上記明細データに挿入するステップを設けたので、上記識別記号をソートキーとしてソートすることができるので、より高速に上位層の集計項目データについての集計値を得ることができる。

【００４６】また、集計項目データから集計対象とする集計項目データを選択するステップを設けたので、集計項目データより上位層又は下位層の集計項目データについての集計値をより高速に得ることができる。

【００４７】さらにまた、階層構造情報テーブルには、階層ごとのレベル名、そのレベルに属する要素名、それらの要素間の順序関係を含むように構成したので、集計項目データより上位層又は下位層の集計項目データについての集計値を得ることができるとともに、要素間の順序関係により集計のレベルを設定することができる。

【００４８】また、明細データを任意の選択による上記集計項目データを含む全ての組み合わせごとに集計するように構成したので、すべての検索条件に対応することができる。

【００４９】さらに、集計値がゼロのときはゼロデータを集計値として保持するように構成したので、集計値がゼロであることを容易に把握することができる。

【図面の簡単な説明】

【図１】　この発明の実施の形態１を示す明細ＤＢシステムの構成図である。

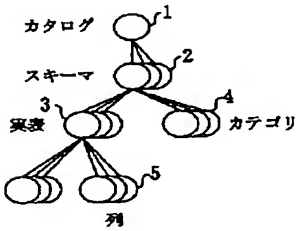
【図２】　この発明の実施の形態１を示すカテゴリの構造図である。

【図３】　この発明の実施の形態１を示す多次元処理の概要説明図である。

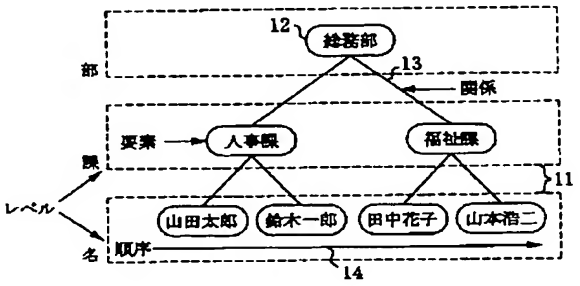
【図４】　この発明の実施の形態１を示すカテゴリと実

表との関連を示す説明図である。
【図５】　この発明の実施の形態１を示す列と仮想的な列の構成図である。
【図６】　この発明の実施の形態１を示す列と仮想的な列との関係を説明する説明図である。
【図７】　この発明の実施の形態１を示すＤＡＴＥ型の階層構造を表す構造図である。
【図８】　この発明の実施の形態２を示す集計方法の相違を示す比較図である。
【図９】　この発明の実施の形態２を示すカテゴリの内容を示す構成図である。
【図１０】　この発明の実施の形態２を示す明細データファイルの構造図である。
【図１１】　この発明の実施の形態２を示すＳＱＬ文のリストである。
【図１２】　この発明の実施の形態２を示す３種類の集計結果を示す出力リストである。
【図１３】　この発明の実施の形態２を示すゼロ集計の説明図である。
【図１４】　この発明の実施の形態３を示すデータフロー図である。
【図１５】　この発明の実施の形態３を示すフローチャートである。
【符号の説明】
３　実表（明細データ）、４　カテゴリ（階層構造情報テーブル）、１２　要素。

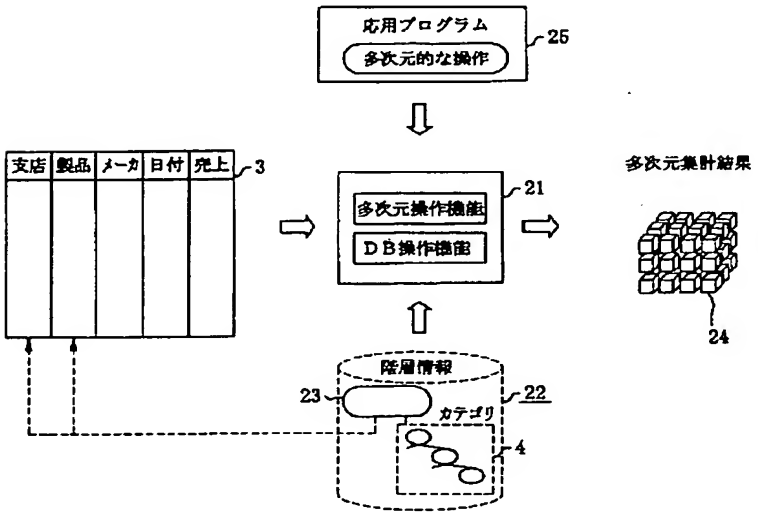
【図１】



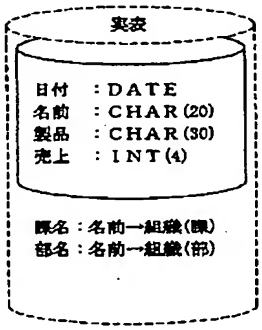
【図２】



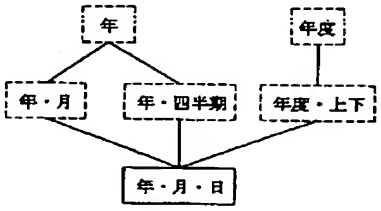
【図３】



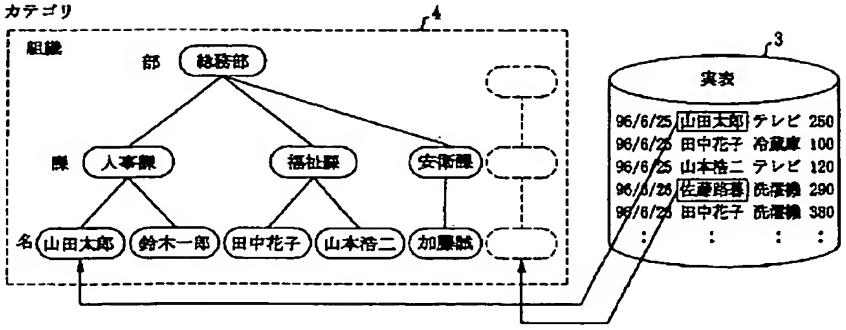
【図５】



【図７】



【図4】



【図8】

(a)

[GROUP BY]

(＊、＊、＊)

(b)

[ROLLUP]

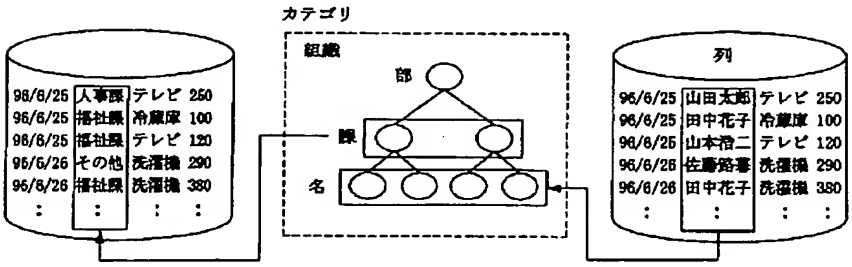
(＊、＊、＊)
(＊、＊、ALL)
(＊、ALL、ALL)
(ALL、ALL、ALL)

(c)

[CUBE]

(＊、＊、＊)
(＊、＊、ALL)
(＊、ALL、＊)
(＊、ALL、ALL)
(ALL、＊、＊)
(ALL、＊、ALL)
(ALL、ALL、＊)
(ALL、ALL、ALL)

【図6】



【図9】

年　月　日	支店	支部	メーカ	製品名	分野
(暗黙の関係)	片瀬	鎌沢	(関係は未導入)	テレビ	AV
	鵠沼			ビデオ	
	大船	鎌倉	A社	洗濯機	家電
	町屋		B社	冷蔵庫	

【図10】

明細データファイル

年月日	支店	メーカ	製品名	売値
98/1/3	片瀬	A社	テレビ	62800円
98/1/3	鵠沼	A社	冷蔵庫	49500円
98/1/3	町屋	B社	洗濯機	78800円
96/1/3	片瀬	A社	冷蔵庫	67800円
96/1/3	片瀬	B社	ビデオ	37000円
96/1/3	町屋	B社	冷蔵庫	88300円
96/1/3	片瀬	B社	ビデオ	47700円
98/1/3	大船	A社	テレビ	62100円
96/1/3	町屋	A社	ビデオ	38900円
96/1/3	鵠沼	B社	テレビ	61900円
96/1/3	片瀬	A社	洗濯機	75800円
96/1/3	大船	B社	冷蔵庫	56600円
：	：	：	：	：

【図11】

片瀬支店　月別　分野別　売上表(1)

```
SELECT      月、分野、製品、COUNT(売値)、SUM(売値)
FROM        実表
WHERE       販売支店='片瀬' AND EXTRACT (MONTH FROM 日付) IN (1,2,3)
GROUP BY    EXTRACT (MONTH FROM 日付) AS 月、
            CATEGORY (売上製品 TO 製品一覧、分野名) AS 分野、
            CATEGORY (売上製品 TO 製品一覧、製品名) AS 製品
```

片瀬支店　月別　分野別　売上表(2)

```
SELECT      月、分野、製品、COUNT(売値)、SUM(売値)
FROM        実表
WHERE       販売支店='片瀬' AND EXTRACT (MONTH FROM 日付) IN (1,2,3)
GROUP BY    EXTRACT (MONTH FROM 日付) AS 月、
            CATEGORY (売上製品 TO 製品一覧、分野名) AS 分野、
            CATEGORY (売上製品 TO 製品一覧、製品名) AS 製品
WITH ROLLUP
```

片瀬支店　月別　分野別　売上表(3)

```
SELECT      月、分野、製品、COUNT(売値)、SUM(売値)
FROM        実表
WHERE       販売支店='片瀬' AND EXTRACT (MONTH FROM 日付) IN (1,2,3)
GROUP BY    EXTRACT (MONTH FROM 日付) AS 月、
            (CATEGORY (売上製品 TO 製品一覧、分野名) AS 分野、
             CATEGORY (売上製品 TO 製品一覧、製品名) AS 製品)
WITH CUBE
```

【図12】

(a)

片瀬支店	月別	分野別	売上表(1)
1月	AV	テレビ 21台	1356800円
		ビデオ 15台	980100円
	家電	洗濯機 6台	477000円
		冷蔵庫 10台	1032500円
2月	AV	テレビ 19台	1118500円
		ビデオ 11台	751800円
	家電	洗濯機 6台	429100円
		冷蔵庫 8台	744500円
3月	AV	テレビ 25台	1952200円
		ビデオ 23台	1660400円
	家電	洗濯機 11台	876600円
		冷蔵庫 16台	1400500円

(b)

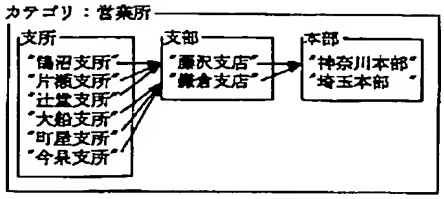
片瀬支店	月別	分野別	売上表(2)
1月	AV	テレビ 21台	1356800円
		ビデオ 15台	980100円
		小計 36台	2336900円
	家電	洗濯機 6台	477000円
		冷蔵庫 10台	1032500円
		小計 16台	1509500円
	中計		52台 3846400円
2月	AV	テレビ 19台	1118500円
		ビデオ 11台	751800円
		小計 30台	1870300円
	家電	洗濯機 6台	429100円
		冷蔵庫 8台	744500円
		小計 14台	1173600円
	中計		44台 3043900円
3月	AV	テレビ 25台	1952200円
		ビデオ 23台	1660400円
		小計 48台	3612600円
	家電	洗濯機 11台	876600円
		冷蔵庫 16台	1400500円
		小計 27台	2277100円
	中計		75台 5889700円
合計			171台 12780000円

(c)

片瀬支店	月別	分野別	売上表(3)
1月	AV	テレビ 21台	1356800円
		ビデオ 15台	980100円
		小計 36台	2336900円
	家電	洗濯機 6台	477000円
		冷蔵庫 10台	1032500円
		小計 16台	1509500円
	中計		52台 3846400円
2月	AV	テレビ 19台	1116500円
		ビデオ 11台	751800円
		小計 30台	1870300円
	家電	洗濯機 6台	429100円
		冷蔵庫 8台	744500円
		小計 14台	1173600円
	中計		44台 3043900円
3月	AV	テレビ 25台	1952200円
		ビデオ 23台	1660400円
		小計 48台	3612600円
	家電	洗濯機 11台	876600円
		冷蔵庫 16台	1400500円
		小計 27台	2277100円
	中計		75台 5889700円
1Q	AV	テレビ 65台	4427500円
		ビデオ 49台	3392300円
		小計 114台	7819800円
	家電	洗濯機 23台	1782700円
		冷蔵庫 34台	3177500円
		小計 57台	4960200円
合計			171台 12780000円

【図13】

<使用するカテゴリ>



<SQL文>

```
SELECT 支部、支所、SUM(売上げ) AS"売上げ高"
  FROM 売上げDB WHERE 本部 = "神奈川本部"
 GROUP BY CATEGORY (支所名 TO 営業所. 支部) AS 支部、
          CATEGORY (支所名 TO 営業所. 支所) AS 支所
  WITH ROLLUP [SUM_ZERO]
```

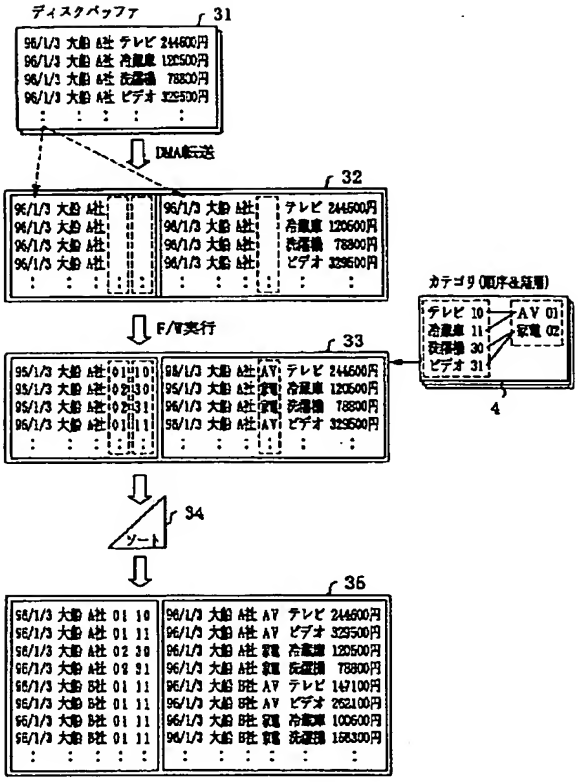
<結果> ゼロ集計を指定しない場合

支部	支所	売上げ高
藤沢支部	片瀬支所	900000
藤沢支部	鵠沼支所	560000
藤沢支部	ALL	1460000
鎌倉支部	大船支所	880000
鎌倉支部	町屋支所	950000
鎌倉支部	ALL	1830000
ALL	ALL	3290000

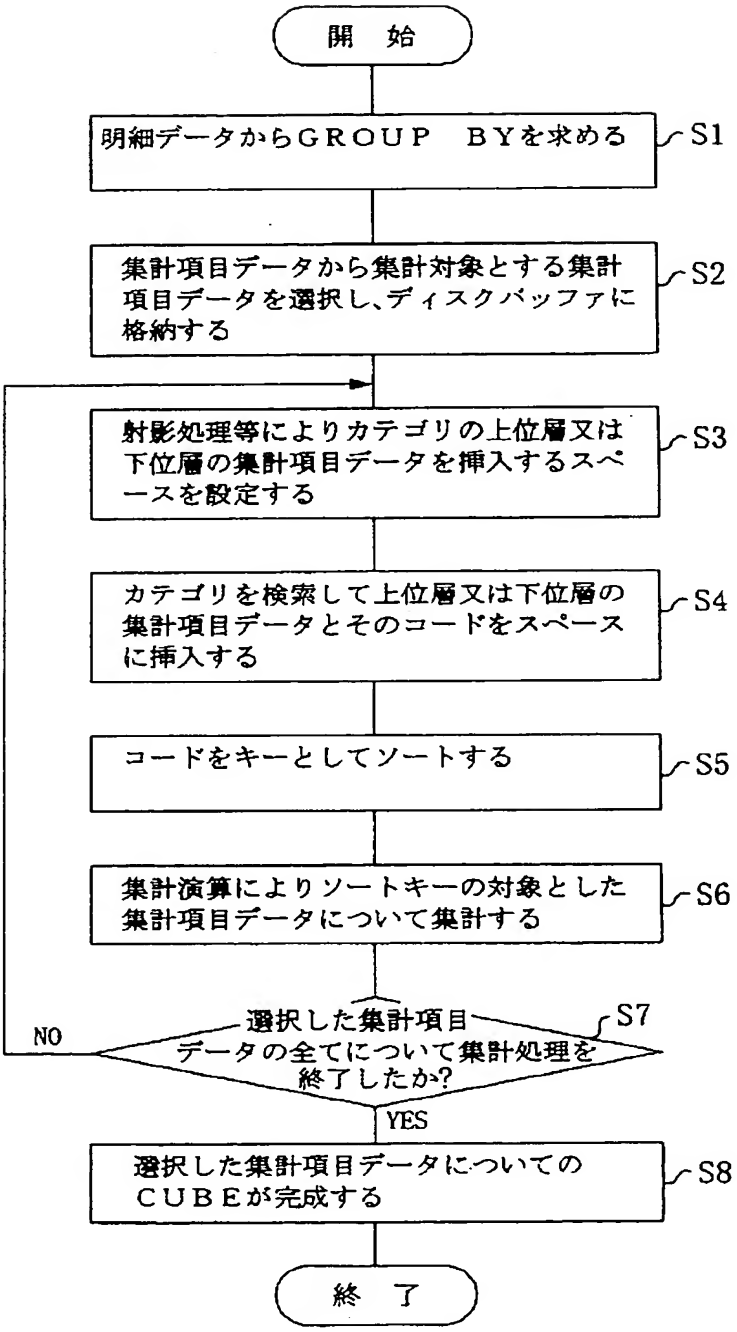
ゼロ集計を指定した場合

支部	支所	売上げ高	
藤沢支部	片瀬支所	900000	
藤沢支部	鵠沼支所	560000	
藤沢支部	辻堂支所	0	←追加
藤沢支部	ALL	1460000	
鎌倉支部	大船支所	880000	
鎌倉支部	町屋支所	950000	
鎌倉支部	ALL	1830000	
鎌倉支部	今泉支所	0	←追加
鎌倉支部	ALL	1830000	
ALL	ALL	3290000	

【図14】



【図15】


開　始
明細データからＧＲＯＵＰ　ＢＹを求める
S1
集計項目データから集計対象とする集計項目データを選択し、ディスクバッファに格納する
S2
射影処理等によりカテゴリの上位層又は下位層の集計項目データを挿入するスペースを設定する
S3
カテゴリを検索して上位層又は下位層の集計項目データとそのコードをスペースに挿入する
S4
コードをキーとしてソートする
S5
集計演算によりソートキーの対象とした集計項目データについて集計する
S6
選択した集計項目データの全てについて集計処理を終了したか?
S7
NO
YES
選択した集計項目データについてのＣＵＢＥが完成する
S8
終　了